Demand-driven Delivery Staff Rostering: Preliminary Results

Andrea Rendl and Christina Burt
Satalia

Workshop on Modelling and Reformulation ModRef-2018
Context of this work: Delivery company

- Company sells goods that require a **manual setup**
- Company delivers with their **own fleet and staff**
- Customers **select** delivery date and time window
Cyclic Roster for Drivers

week 1

shift pattern 1

shift pattern 2

week 2
Van hours and resulting delivery capacity
Problem: capacity from roster does not match demand

Possible deliveries versus real demand

- van-hours
- possible deliveries
- real demand
Our goal: find a roster that matches the real demand
Mathematical Model
Parameters

- Shift patterns (weeks) $S$
- Drivers / vans $V$
- Estimated demand per weekday (in orders) $O$
Constants (1/2)

- Time factor $\tau$
- Time units $T = \{1.. 24*\tau\}$
- Stem time $t_{\text{stem}}$
- Lunch break duration $t_{\text{lunch}}$
Constants (2/2)

- Max working hours
  - $t_{\text{daily}}$
  - $t_{\text{weekly}}$

- Paid working hours $t_{\text{paid}}$

- Shift constants
  - min/max shift length
  - Earliest start time
  - Latest end time

- $o$ in $\mathbb{R}^+$
  average orders delivered per van per hour

- $vv^s_v$ in $\{0,1\}$
  1 if van $v$ is assigned to shift pattern $s$
Main Decision variables

- $s_{d}^{s}$ in T
  Start time of shift on weekday d, for shift pattern s

- $e_{d}^{s}$ in T
  End time of shift on weekday d, for shift pattern s
Helper Decision variables

- $l^s_d$ in $T$
  length of shift on weekday $d$, for shift pattern $s$

- $w^s_d$ in $\{0, 1\}$
  1 if weekday $d$ in shift pattern $s$ is a working day

- $vh_d$ in $\{0 .. T_{max}\}$
  The number of hours all vans are working on weekday $d$
“Objective” decision variables

- \( a_d \) in \( \mathbb{R}^+ \)
  The average number of orders delivered on weekday \( d \) (over all shift patterns)

- \( u_d \) in \( \mathbb{R}^+ \)
  Unmet demand (in orders) on weekday \( d \), over all shift patterns
Shift Constraints

- \( s^s_d \geq \text{earliestStartTime} \)  \( \forall s,d \)
- \( e^s_d \leq \text{latestEndTime} \)  \( \forall s,d \)
- \( e^s_d \geq s^s_d \)  \( \forall s,d \)
- \( l^s_d = e^s_d - s^s_d \)  \( \forall s,d \)
- \( l^s_d \leq M \times w^s_d \)  \( \forall s,d \) with \( M \geq t_{\text{day}} \)
Working hour Constraints

- \[ \sum_{s,d} \{ l^s_d \} - \sum_{s,d} \{ w^s_d \cdot t_{\text{lunch}} \} = t_{\text{paid}} \]
  The average number of working hours over all shift patterns must be equal to the number of paid hours.

- \[ \sum_{d} \{ l^s_d \} - \sum_{d} \{ w^s_d \cdot t_{\text{lunch}} \} \leq t_{\text{week}} \quad \forall s \in S \]
  For each shift pattern, the maximal number of working hours is not exceeded.
2-day break Constraints

- \((w^s_{\text{Sat}} + w^s_{\text{Sun}} = 0) + (w^s_{\text{Sun}} + w^{s+1}_{\text{Mon}} = 0)\)
  
  \[+ (w^s_{\text{Mon}} + w^s_{\text{Tue}} = 0) = 1 \quad \forall \ s \in S - 1\]

- \((w^S_{\text{Sat}} + w^S_{\text{Sun}} = 0) + (w^S_{\text{Sun}} + w^1_{\text{Mon}} = 0)\)
  
  \[+ (w^S_{\text{Mon}} + w^S_{\text{Tue}} = 0) = 1\]

There is a two day break between each shift
Van hour Constraints

- \( vh_d = \sum_{s,v} \{ vv^s_v \cdot l^s_d \} \)

- \( - \sum_{s,d} \{ w^s_d \} \cdot \sum_{s,v} \{ vv^s_v \cdot t_{lunch} \} \) \( \forall d \)

Calculating the van hours \( vh_d \) for each weekday \( d \), over all shift patterns

- \( vv^s_v \in \{0,1\}: 1 \) if van \( v \) is assigned to shift pattern \( s \) (constant)
- \( w^s_d \in \{0,1\}: 1 \) if weekday \( d \) in shift pattern \( s \) is a working day
Serviced-orders Constraints

\[ a_d = o \times (v_h_d - 2 \times t_{stem}) \times \{ \sum_{s,d} w^s_d \} \times \sum_{s,v} \{ v_v^s \} \] \quad \forall d

Calculating the average number of serviced orders (fleet capacity) \( a_d \) for each weekday \( d \): multiplying \( o \) with the net worked hours (removing the stem time)

- \( o \): average orders delivered per van per hour
- \( v_v^s \) in \( \{0,1\} \): 1 if van \( v \) is assigned to shift pattern \( s \) (constant)
- \( w^s_d \) in \( \{0, 1\} \): 1 if weekday \( d \) in shift pattern \( s \) is a working day
Unmet demand Constraints

- \( u_d = |O_d - a_d | \) \( \forall d \)

The unmet demand \( u_d \): the absolute value of expected order \( O_d \) minus the fleet capacity \( a_d \)

- \( O_d \) in \( \mathbb{R} \): expected number of orders on day \( d \)
- \( a_d \) in \( \mathbb{R} \): average fleet capacity in number of orders
Objective 1: minimize unmet demand

- Minimize $p$

Minimize the maximal unmet demand $p$

- $p \geq 0.0$
- $p \leq \text{max demand}$
- $p > u_d \quad \forall \ d$
Objective 2: weighted unmet demand

- Minimize $\sum_d \{ c_d \times u_d \}$

Minimize the unmet demand $u_d$ weighted with $c_d$
Preliminary Results
MiniZinc model

- Implemented model in MiniZinc
- Model + data available on github (MIT license):
  https://github.com/angee/demand-shift-pattern

(link is also in the paper)
Problem instances

● Parameters:
  ○ Vans/drivers: 12, 24, 60
  ○ Shift patterns: 2, 4, 6
  ○ 2 Demand scenarios:
    ■ Linear-increase of demand over week
    ■ Peak demand on Thu/Fri

● Reflect real-world problem sizes
Experimental Setup

- MiniZinc v2.1.7
- Solvers:
  - Gecode
  - COIN-OR cbc
- Timeout: 300 seconds
- Default search
<table>
<thead>
<tr>
<th>Instance</th>
<th>Runtime (sec)</th>
<th>Objective</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Gecode</td>
<td>cbc</td>
</tr>
<tr>
<td>v12 s6 linear</td>
<td>300.000</td>
<td>300.000</td>
</tr>
<tr>
<td>v24 s2 linear</td>
<td>0.027</td>
<td>0.353</td>
</tr>
<tr>
<td>v24 s4 linear</td>
<td>300.000</td>
<td>10.006</td>
</tr>
<tr>
<td>v60 s6 linear</td>
<td>300.000</td>
<td>241.377</td>
</tr>
<tr>
<td>v60 s2 linear</td>
<td>0.026</td>
<td>0.328</td>
</tr>
<tr>
<td>v60 s4 linear</td>
<td>300.000</td>
<td>11.444</td>
</tr>
<tr>
<td>v24 s6 linear</td>
<td>300.000</td>
<td>300.000</td>
</tr>
<tr>
<td>v12 s2 linear</td>
<td>300.000</td>
<td>0.317</td>
</tr>
<tr>
<td>v12 s4 linear</td>
<td>300.000</td>
<td>17.668</td>
</tr>
<tr>
<td>v60 s4 peak-thu-fri</td>
<td>300.000</td>
<td>2.404</td>
</tr>
<tr>
<td>v12 s2 peak-thu-fri</td>
<td>0.013</td>
<td>0.292</td>
</tr>
<tr>
<td>v60 s2 peak-thu-fri</td>
<td>0.014</td>
<td>0.327</td>
</tr>
<tr>
<td>v12 s4 peak-thu-fri</td>
<td>300.000</td>
<td>2.161</td>
</tr>
<tr>
<td>v24 s2 peak-thu-fri</td>
<td>0.047</td>
<td>0.284</td>
</tr>
<tr>
<td>v60 s6 peak-thu-fri</td>
<td>300.000</td>
<td>0.728</td>
</tr>
<tr>
<td>v12 s6 peak-thu-fri</td>
<td>300.000</td>
<td>1.16</td>
</tr>
<tr>
<td>v24 s4 peak-thu-fri</td>
<td>300.000</td>
<td>5.765</td>
</tr>
<tr>
<td>v24 s6 peak-thu-fri</td>
<td>300.000</td>
<td>0.962</td>
</tr>
</tbody>
</table>
Observations

- MIP solver outperforms CP solver
  - We do not use full power of CP
    - search strategy
    - global constraints

- Several optimal solutions cannot match demand
  - Working hour settings very conservative
Future Work

- Alternative **CP-style** formulation
  - Global constraints
  - Custom search strategies

- Include **optional constraints**
  - E.g. holidays every other Saturday

- Evaluate **constant settings**: with what settings can we find a solution to fully match the demand?